

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-287697

(43)Date of publication of application : 01.11.1996

-----

(51)Int.Cl. G11C 16/06

G06F 12/16

G06F 12/16

-----

(21)Application number : 07-117920 (71)Applicant : NIPPONDENSO CO LTD

(22)Date of filing : 18.04.1995 (72)Inventor : KUREBAYASHI TAKESHI

---

(54) MEMORY DEVICE

(57)Abstract:

PURPOSE: To provide a memory device which gives reliable information even beyond a rewritable limit by dividing the memory area of a memory device into plural blocks and executing required writing to a block selected different from the previous time in order.

CONSTITUTION: Information and a group of data required are stored in advance into an EEPROM10 connected to the CPU 2 in an internal combustion engine control device. A control device 1 compares the code sent from a vehicle theft preventing device 9 with the one stored in the EEPROM10. When these are not mutually coincident, it is judged that start-up due to abnormal operation has occurred and the drive of an injector 8 is suspended. When the codes are mutually coincident, it is judged that the starting operation is normal and the vehicle theft preventing device 9 is notified of the coincidence. Again, a different new code is sent from the vehicle theft preventing device 9 to the control device 1 to be stored by the EEPROM. A new code is rewritten into the EEPROM10 at every normal start-up.

**\* NOTICES \***

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.\*\*\*\* shows the word which can not be translated.

3.In the drawings, any words are not translated.

---

**CLAIMS**

---

[Claim(s)]

[Claim 1]A storage device which has and reads a nonvolatile memory which has the rewriting restricted frequency which is provided with the following and characterized by updating the aforementioned block which can be read, or said block which can be written in one by one and with which writing is carried out by a program.

A data storage means divided into two or more blocks used as one read-out of

data and one unit of writing.

A state storing means which memorizes any at least one of the referred data in which a state to that effect is shown at the time of a write-in end or a write-in start to said block each.

A block decision means which judges a block which can be read, and its following block using said referred data for a block in which writing of data carried out normal termination to the last to be the block which can be written in.

A read-out writing means which read data to the aforementioned block which can be read, or data is written in to said block which can be written in, and updates the referred data in said state storing means at the time of said write-in start or a write-in end.

[Claim 2]The storage device according to claim 1 updating said referred data in which said state storing means memorizes referred data in which a write-in start state and write-in exit status are shown, respectively, the aforementioned read-out writing means is written at the time of a write-in start, and a start is shown, and said referred data in which a write-in end is shown at the time of a write-in end.

---

## DETAILED DESCRIPTION

---

[Detailed Description of the Invention]

[0001]

[Industrial Application] Like the rolling code of learnt values, such as control, or a vehicle theft arrester, it is updated one by one, or renewal of the number of times of many is carried out dramatically, and this invention relates to the storage device for memorizing the data which should always be held.

[0002]

[Description of the Prior Art] Even if electric supply is intercepted, a memory content can be held, and EEPROM is used as a memory which can rewrite the contents. Generally, EEPROM is the memory to which the number of times of rewriting was restricted, and the upper limit (rewriting restricted frequency) of the number of times of rewriting per byte is 10000. It is a time grade. For this reason, in the computer systems which have a storage device using EEPROM, it is designed so that it may rewrite in the life cycle of that system and the number of times may not exceed this upper limit. However, it is to rewrite exceeding this upper limit depending on the purpose of use. Therefore, in order to correspond in such a case, the storage area of EEPROM to carry is divided into two or more blocks, and when the writing to a certain block exceeds prescribed frequency,

there is art of the storage device written in other blocks. In rewriting for every block, in spite of not exceeding upper limit as the number of times of rewriting of each data within a block, In order to rewrite as the whole block, upper limit may be exceeded, and since it is not efficient, there is also memory technology in which it rewrites for every data and the counter for frequency recording is formed.

[0003]

[Problem(s) to be Solved by the Invention]However, when two or more data with relation is written in one field of a memory also in the system of which the above-mentioned rewriting, When electric supply is intercepted before rewriting of the data of a series of was completed, the portion in which data was changed in the one field as a result, and the portion which is not changed exist. For this reason, when it energizes again, even if it reads a series of data of that field, there is a problem that those data is not exact. Since data is written in the always same field even if judged with interception of electric supply in the middle of write-in [ of data ], the data memorized at the end among proper data cannot be read, either. Especially, in the system of the latter rewriting, although it can certainly use to the number of times of rewriting, the half of a memory will be occupied at a counter and there is a problem that the utilization efficiency as a memory is bad.

[0004]Therefore, it aims at holding and reading the data which the last normal

storage operation completed, and being made to be made, even if faults, such as write-in discontinuation, occur in this invention in view of the above-mentioned technical problem.

[0005]

[Means for Solving the Problem] In a storage device with which composition of this invention has a nonvolatile memory which has rewriting restricted frequency, it reads, and writing is carried out by a program in order to solve the above-mentioned technical problem, A data storage means divided into two or more blocks used as one read-out of data and one unit of writing, A state storing means which memorizes any at least one of the referred data in which a state to that effect is shown at the time of a write-in end or a write-in start to said block each, A block decision means which judges a block which can be read, and its following block using said referred data for a block in which writing of data carried out normal termination to the last to be the block which can be written in, Read data to the aforementioned block which can be read, write in data to said block which can be written in, and. It is having a read-out writing means which updates the referred data in said state storing means at the time of said write-in start or a write-in end, and updating the aforementioned block which can be read, or said block which can be written in one by one. As for composition of related inventions, said state storing means memorizes referred data in which a write-in

start state and write-in exit status are shown, respectively, and the aforementioned read-out writing means is updating said referred data in which it writes at the time of a write-in start, and a start is shown, and said referred data in which a write-in end is shown at the time of a write-in end.

[0006]

[Function]The writing of the data which divides into two or more blocks the storage area of the nonvolatile memory to which the number of times of rewriting was restricted, for example, EEPROM, and needs it is carried out to the block which was chosen in order unlike last time. Selection of the block which can be read, and the block which can be written in is performed by referring to the referred data in which the state of a write-in end or a write-in start where it corresponded for every block is shown and which was memorized by the state storing means.

[0007]When it writes in a state storing means and only the state of the end is memorized, referred data is investigated in order by a block decision means, the block which writing completed last time is judged, and the block which can be read, and its following block are determined for the block as the block which can be written in. Since all block is not written in as for first-time writing, either, the block of No. 1 is applicable. When the writing of data is completed, the referred data value which writes in the state storing means corresponding to the block



which performed the writing, and means an end is made to memorize. Even if electric supply is intercepted, rewriting processing stops in the middle of data rewriting and data rewriting is imperfectly completed to it, Since the block of rewriting discontinuation is written in and the referred data of the end is not set up when a power supply is recovered and refers to the rewritten data again after that, the previous block is recognized to have memorized latest data.

[0008]When it writes in a state storing means and only the state of the start is memorized, referred data is investigated in order by a block decision means, the block with which writing was started last time is judged, and the block which can be written in, and the previous block are determined for this block as the block which can be read. In this case, if electric supply is intercepted and rewriting processing is suspended in the middle of data rewriting, after that, a power supply is recovered, and since it rewrites when referring to the rewritten data, and the judgment of discontinuation cannot be performed, the block of the front before that will be again recognized to have memorized positive data.

[0009]Are a case where write in a state storing means with a write-in end, and the state of the start is memorized in the composition of claim 2, and by a block decision means. The block which investigated the state of the write-in end and writing completed last time is judged, and the block which can be read, and its following block are determined for the block as the block which can be written in.

A state with the block which writes in with the block used as a write-in end, and is started is compared, and it is judged [ or or ] by that writing was interrupted last time whether was completed. Even if a power supply is intercepted, rewriting processing stops in the middle of data rewriting and data rewriting is imperfectly completed to it, Since the block of rewriting discontinuation is written in and the referred data of the end is not set up when a power supply is recovered and refers to the rewritten data again after that, the previous block is recognized to have memorized latest data.

[0010]

[Effect of the Invention]It is made the nonvolatile memory to which the number of times of rewriting was restricted, for example, the composition which uses the storage area of EEPROM for a block dividing it, and rewrites in writing in for every block, and the number of times increases seemingly. Since the referred data of a write-in end or a write-in start in which either is shown at least is provided and referred to, even when writing is interrupted and completed on the way, the data properly memorized at the end before discontinuation can be used after resumption. As the actual condition of the block the block which writes in is not only determined in the composition of claim 2, but, it ended by the suspended state -- what can be written in a sake, and the thing which can be written in where the pre- block was written in normally and completed can be

distinguished, and the duplicate writing can be avoided, or it can be made to take processing so that fault may not occur in collation of a code, etc.

[0011]

[Example] Hereafter, this invention is explained based on a concrete example.

Drawing 1 is a block lineblock diagram of the internal-combustion engine control system 100 which used the storage device of this invention for memory of the code (called a rolling code) shared between the combustion engine control device 1 and the vehicle theft arrester 9. In drawing 1, EEPROM which is the memory to which the number of times of rewriting was restricted is displayed as E<sup>2</sup>PROM.

[0012] The combustion engine control device 1 comprises a microcomputer. The filter 6 which inputs the inlet-pipe internal pressure signal S2 by the intake pressure sensor 4 which the waveform shaping circuit 5 which inputs the rotation signal S1 from the rotation sensor 3 formed in the internal-combustion engine (not shown) was connected, and was formed in the inlet pipe of the internal-combustion engine is connected to CPU2.

[0013] The control program is memorized by the program memory (ROM) 7 connected with CPU2, and control of an internal-combustion engine is performed because CPU2 executes this control program. Each aforementioned signal S1 and S2 are used with a control program, and the injector 8 of an

internal-combustion engine is controlled as an SPI (Single Point Injection) function here. In addition, although not illustrated, the A/D converter and D/A conversion which were connected to CPU2 are controlled, and the signal which is needed for control is exchanged.

[0014]CPU2 of the combustion engine control device 1 is connected with the vehicle theft arrester 9. The vehicle theft arrester 9 supervises whether the internal-combustion engine started by normal starting operation. From the vehicle theft arrester 9, the code beforehand memorized in the device corresponding to the normal starting operation at the time of start up is transmitted to the combustion engine control device 1 at every start up of an internal-combustion engine. EEPROM10 by which the code is connected to CPU2 in the combustion engine control device 1 -- the information which needs others beforehand -- a group -- it memorizes as data. It is used dividing an inside into N blocks so that it may mention later, and EEPROM10 is written in as a state storing means referred data is remembered to be, it writes in with the start table 11, and the end table 12 is formed in EEPROM10.

[0015]As compared with the code which has memorized to EEPROM10 the code sent from the vehicle theft arrester 9, the combustion engine control device 1 is judged to be start up of the internal-combustion engine by abnormal operation, when these are not in agreement, and it stops the drive of the injector

8. Since it is judged as normal starting operation when a code is in agreement, the vehicle theft arrester 9 is notified that it was in agreement. and -- a new code which is different from the vehicle theft arrester 9 again before corresponding to the starting operation of an internal-combustion engine is transmitted to the combustion engine control device 1 -- in addition, required information -- a group -- EEPROM10 memorizes as data. Thus, EEPROM10 is rewritten every, whenever starting operation is performed normally (reading writing means). The vehicle theft arrester 9 has a function sent out in order to also make the code of 1 time ago hold and compare as a fail safe function at the time of rewriting.

[0016]The writing of EEPROM10 is as follows. From the physical limit of EEPROM10 the very thing, it is usually 10000. The rewriting restricted frequency about a time is set up from a viewpoint of a guarantee of stored data. When making this into the maximum use count Nmax as a vehicle theft arrester on the other hand supposing the maximum number of times of rewriting of the code over the employment period of vehicles, for example, ten years, and 15 years, The code storage area of EEPROM can be divided into N blocks shown by the following one formula, and a possibility of exceeding the rewriting restricted frequency of EEPROM can be made low by making these blocks rewrite and memorize a code one by one and cyclically.

[Number 1] 
$$N_{\max} / 10000 \leq N$$
 [0017]The writing and read-out to EEPROM10

are carried out by 1 block unit (data storage means). And the block which writes in is changed one by one for every one writing so that the last latest data (information containing a code) may not be overwritten. Therefore, even when writing is interrupted, the latest data which was written in last time before it and completed is always left behind.

[0018] So that it may be checked later whether the writing to each N blocks was completed normally, or writing has been started, It writes in with the write-in start table 11 which makes the referred data in which those states are shown at the time of a write-in start and a write-in end correspond to each block each, and makes it memorize, and the end table 12 is prepared. Since the table value (referred data) which writes in with this write-in start table 11, and is memorized on the end table 12 needs to be memorized also after power OFF, it is memorized by EEPROM10. The write-in write-in start end table 11 and 12 is a table which can memorize the value for N pieces as shown in drawing 3. The value memorized to each address of each table shows the write-in start state and the write-in exit status of the block corresponding to the address. Therefore, each of these tables 11 and 12 do not need to be taken in the memory area which not necessarily continued, and they may be provided so that it may belong to a part of each block area of EEPROM10. Or this table region is established in the lot of EEPROM10, and it may be made to divide the remaining fields into N

blocks.

[0019]The block with which latest data is memorized and which can be read, and the block which should be written in the next and which can be written in are judged because CPU2 investigates these table values (block decision means).

The block decision means of this CPU2 is included in the control program memorized by the program memory 7.

[0020]Rewriting has N pieces (N which fills several 1 formula) of the storage area of EEPROM10 blocked in order from the 1st to the Nth now here. and -- writing is completed to the Xth of N blocks -- this Xth block to a group -- a group which data was read, and the code was read and contained a new code in the X+1st following blocks after the appropriate anti-theft judging -- suppose that data is written in. At the stage before being written in the X+1st blocks, it is drawing 3 (a) in the write-in start table 11. Since writing is already carried out to the 1st to each Xth block so that it may be shown, the table value of Y+1 which shows that the write-in start was already carried out is memorized. And the table value over the block of the X+1st henceforth means that the value Y is memorized and writing has not been carried out yet. That is, the memory content of the write-in start table 11 is rewritten when the block is used as writing, and the value which added 1 to the value (here Y) memorized now begins writing.

[0021]When the combustion engine control device 1 receives a code from the

vehicle theft arrester 9 by the case of being above, at the time of start up, A code is read from the last latest data memorized to EEPROM10, normal abnormality judgement of key operation is performed, and the procedure of receiving the still newer code in proper start up, and memorizing new data to EEPROM10 is explained using the flow chart of drawing 2.

[0022](1) At Step 202, it is investigated first a block of what position the block which should read and which can be read is. If the block which can be read is determined, the block which writes in the next and which can be written in will turn into a block of the 1st [ +] number. The write-in end table 12 is referred to, the table value is searched in order and the adjoining table value is compared. The X+1st blocks used as the value which differs in those values are determined as the block which has not been rewritten yet, i.e., the block which can be written in, and it is decided that it will be the block with which the Xth block is rewritten last time and latest data is left behind, i.e., the block which can be read. Y+1 is written in the table value corresponding to the Xth block written in at the end here, and Y is written in the table value corresponding to the X+1st following blocks (drawing 3 (a)).

[0023](2) Carry out reception of the code which processing which reads the data memorized by the Xth block that should be read is carried out, and is generated by starting operation from the vehicle theft arrester 9 at Step 204 being sent. And



at Step 206, the memorized code is compared with the code sent from the vehicle theft arrester 9, and a theft judging is carried out.

(3) Since it will be supposed as a result of code comparison of Step 206 that it is unusual start up if the code is inharmonious, in order to receive a new code, not to rememorize and to carry out another processing, end this flow. If the code is in agreement, it will be supposed that it is normal start up and it will progress to the following step. Code comparison when writing is interrupted is also actually included in the code comparison here.

(4) At Step 208, a normal thing is first notified to the vehicle theft arrester 9. And since a new code is sent out from the vehicle theft arrester 9 to the combustion engine control device 1, reception of the new code is carried out.

[0024](5) In the combustion engine control device 1, in order to write in the data based on a new code, table value Y of the X+1st blocks of the write-in start table 11 is updated by Y+1 at Step 210 (drawing 3 (b)). And the data based on the new code sent from the vehicle theft arrester 9 to the X+1st blocks at Step 212 is written in.

(6) If the data processing is completed, at Step 214, the X+1st table values of the write-in end table 12 will be updated from Y to Y+1 (drawing 3 (c)), the information on rewriting having been completed will be set, and processing of rewriting will be completed.

[0025]When writing stops on the way and is not normally completed for a certain reason in the midst of Step 212, The code contained in the data currently written in the X+1st blocks is unreliable, therefore these X+1st data is unnecessary, and may be again written in the X+1st blocks. In order to check whether the last writing is completed on the way, the write-in start table 11 is investigated. Since the X+1st table values are Y, it is checked that these X+1st blocks are not what was stopped on the way.

[0026]When a program is resumed again and the block which can be read, and the block which can be written in are judged at Step 202 after discontinuation, the X+1st memories of the write-in end table 12 in drawing 3 (c) are Y with as. Since the perfect latest data in which writing was ended is memorized by the Xth block, the block which can be read, and the X+1st blocks are judged for the Xth block to be the block which can be written in. When saying that it is interrupted and the updated code is not memorized correctly, It can respond by a code not being in agreement probably and the judgment of being discontinuation accomplishing inside the above-mentioned step 206, and performing code comparison with the code before [ one ] that is notified to the vehicle theft arrester 9, makes the code of 1 time ago call and is contained in latest data, if it is discontinuation. Therefore, even when processing is interrupted on the way, the right data to need is read without needing a certain post-processing, and the

writing of new data is carried out to the block with which writing stopped on the way once again satisfactorily.

[0027]When this writing is interrupted and writing is again carried out by the same block, it means that writing was carried out doubly. 10000 which the writing of parenchyma rewrites some day and is restricted frequency when the same phenomenon occurs with the same block several times A time will be exceeded. However, there is little probability that the above special rewriting situations will occur frequently with the same block, since each block is rewritten in order, it rewrites on the average and the number of times increases. Therefore, there is almost no influence on the write-in performance of EEPROM10. The discontinuation generating frequency beforehand expected in a simulation etc. is expected, and it may deduct from rewriting restricted frequency.

[0028]However, by referring to the write-in start table 11, it can be checked whether a pre- block completes normally whether the following block which was judged by reference of the write-in end table 12 and which can be written in finished with discontinuation, and writing is started at all. Then, when judged with it being discontinuation, the X+2nd following blocks can be considered as the block which can be written in so that the number of times of writing may not be increased once. However, the block which can be read is the Xth block.

[0029]In each tables 11 and 12 of a write-in start and a write-in end used as

referred data, If the table value over each block will be set as the same value (drawing 3 Y+1) for every rewriting one by one to 1-N, It can be judged in which block latest data always exists on a boundary with change of a table value, and it is easily judged in it that that order is the block which can be read, and the block which can be written in. Since it refers to such a method, the table value does not need to comprise a numerical value which carries out stepping, the value that the boundary of the written-in block and the block written in from now on is judged may be sufficient, and there should just be at least 1 bit in it.

[0030]In order for each N blocks to rewrite and not to exceed restricted frequency, one counter formed independently is made to \*\*\*\*\* every [ 1 ] for every time of the Nth block being rewritten, and it can do by whether the value of this counter exceeds 10000, and \*\* being investigated. It is because what is necessary is just to count the number of times of rewriting of one of blocks since any block is rewritten in order. Or it may be considered as the counter value which each above-mentioned table value makes \*\*\*\*\*.

[0031]As mentioned above, also with the storage device which enabled it to use the actual number of times of writing exceeding the rewriting restricted frequency of EEPROM10, without forming the means of the special measure against fault, the data written in last time can be held certainly and can be read again.

[0032]When the written-in block is the Nth exactly, the memory of the table which

should be written in the next will return to the 1st. The state of each tables 11 and 12 in the time of writing being completed to the Nth block is drawing 4 (a). It is as being shown. drawing 4 (a) \*\*\*\* -- all the values of the tables 11 and 12 are the Y+1 [ same ], and it becomes nothing changing the data value of all the tables. In the algorithm which detected that a data value had change and has determined the block which can be read, and the block which can be written in, those blocks are not determined in the state. Therefore, in such a case, since it is in the state where all the memories were written in briefly exactly, suppose that the measures with which writing is again carried out from the 1st are taken. [0033]And when writing in the 1st block, in the write-in start table 11. After Y+2 which added 1 to the value Y+1 before writing is written in (drawing 4 (b)) and a new code is written in the 1st block, Y+2 is similarly written in the 1st place of the write-in end table 12 (drawing 4 (c)). These Y+2 may be Y. This supports becoming a value of 0 and 1 automatically, if the memory of a table is 1 bit, and +1 is carried out to a front value. Therefore, for the table for N pieces, what is necessary is just to prepare by 2N bit at least, and many memory areas of EEPROM are not monopolized.

[0034](The second example) Although the case where wrote in EEPROM10 in the case of the first example, wrote in with the start table 11, and two of the end tables 12 were installed was shown, only the case that the write-in end table 12

is installed is explained. When only the write-in end table 12 is installed, The block of the boundary where the table value of the write-in end table 12 is changing with which writing is completed and latest data is memorized, i.e., the block which can be read, It also turns out that the boundary of the block which should be written in from now on, i.e., the block which can be written in, will be shown, and writing is completed safely. In this case, that block cannot investigate whether writing was started or not. That is, distinction of whether it was discontinuation and whether the pre- block wrote in normally and was only completed is not attached. Even in this case, since the processing after discontinuation occurs will be the same as the first example, it is satisfactory. since there is an advantage whose field occupied to EEPROM10 requires only a part for N bit at least in the case of this second example, compared with the case where a counter which is in conventional technology is formed for every memory, the utilization efficiency of a memory is boiled markedly and improves.

---

## DESCRIPTION OF DRAWINGS

---

[Brief Description of the Drawings]

[Drawing 1]The block lineblock diagram of the combustion engine control device

using the storage device of this invention.

[Drawing 2] The flow chart of read-out to the storage device of this invention, and write-in processing.

[Drawing 3] The explanatory view of a write-in start table and a write-in end table.

[Drawing 4] The explanatory view of the special example of write-in processing.

[Description of Notations]

1 Combustion engine control device

7 ROM (a control program, a state storing means, and a block decision means are included)

9 Vehicle theft arrester

10 EEPROM (the block which can be read, and the block which can be written in are included)

11 Write-in start table (referred data)

12 Write-in end table (referred data)

(51)Int.Cl. <sup>8</sup>	識別記号	片内整理番号	F I	技術表示箇所
G 1 1 C 16/06			G 1 1 C 17/00	3 0 9 F
G 0 6 F 12/16	3 1 0	7623-5B	G 0 6 F 12/16	3 1 0 A
	3 4 0	7623-5B		3 4 0 P

審査請求 未請求 請求項の数2 F D (全 8 頁)

(21)出願番号 特願平7-117920

(22)出願日 平成7年(1995)4月18日

(71)出願人 000004260

日本電装株式会社

愛知県刈谷市昭和町1丁目1番地

(72)発明者 紅林 毅

愛知県刈谷市昭和町1丁目1番地 日本電装株式会社内

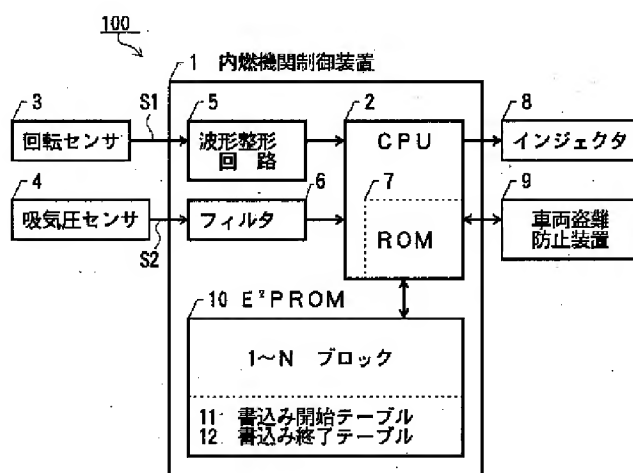
(74)代理人 弁理士 藤谷 修

## (54)【発明の名称】 メモリ装置

## (57)【要約】

【目的】書換え制限回数のある不揮発性メモリを有するメモリ装置を用いて書換え回数を越えて使用する構成で、書込み中断などの不具合が発生しても、確実な情報を保持出来、特別な処理をしなくても良い構成とすること。

【構成】EEPROM10を、必要な書換え回数を書換え制限回数(約10000)で割った値以上の整数N個のブロックに分割して、ブロックへのデータの書込みが書換え毎に順番に実施される。ブロックの選択は、ブロック一つ一つに対して対応させてある書込み終了テーブル12、さらには書込み開始テーブル11が参照されて、読出すブロックと、次に書込むブロックとが決定される。書換え途中で給電が遮断されて、不完全に書換えが終了したとしても、再び電源が回復してデータを参照する際に、中断されたブロックは書込み終了の参照データが設定されていないので、その前のブロックが最新データを記憶していると認識される。





**【特許請求の範囲】**

**【請求項1】** 書換え制限回数を有する不揮発性のメモリを有して読出し書込みがプログラムで実施されるメモリ装置において、

一回のデータの読出しと書込みの一単位となる複数のブロックに分割されたデータ記憶手段と、  
前記ブロック一つ一つに対して、書込終了時に又は書込開始時にその旨の状態を示す参照データの少なくともいずれか一つを記憶する状態記憶手段と、  
前記参照データを用いて、最後にデータの書込みが正常終了したブロックを読出可能ブロック、その次のブロックを書込可能ブロックと判定するブロック判定手段と、  
前記読出可能ブロックに対してデータを読出し、または前記書込可能ブロックに対してデータを書込むと共に、  
前記状態記憶手段において前記書込開始時又は書込終了時に、その参照データを更新する読出書込手段とを有し、  
前記読出可能ブロックまたは前記書込可能ブロックを順次更新することを特徴とするメモリ装置。

**【請求項2】** 前記状態記憶手段は、書込開始状態と書込終了状態とをそれぞれ示す参照データを記憶し、  
前記読出書込手段は、書込開始時に書込み開始を示す前記参照データと、書込み終了時に書込終了を示す前記参照データの更新を行うことを特徴とする請求項1に記載のメモリ装置。

**【発明の詳細な説明】****【0001】**

**【産業上の利用分野】** 本発明は、制御等の学習値や車両盗難防止装置のローリングコード等のように、順次更新され、または非常に多くの回数更新されて、常に保持されているべきデータを記憶するためのメモリ装置に関する。

**【0002】**

**【従来の技術】** 給電が遮断されても記憶内容を保持できて、内容を書き換えることができるメモリとしてEEPROMが利用されている。一般的に、EEPROMは書換え回数が制限されたメモリであって1バイトあたりの書換え回数の上限値（書換え制限回数）は10000回程度である。このためEEPROMを利用したメモリ装置を有するコンピュータシステムでは、そのシステムの製品寿命内において書換え回数がこの上限値を越えないように設計される。しかし使用目的によっては、この上限値を越えて書換えを実施したい場合もある。従ってこのような場合に対応するため、搭載するEEPROMの記憶領域を複数のブロックに分割し、あるブロックへの書き込みが所定回数を越えた時点で他のブロックに書き込むようにしたメモリ装置の技術がある。またブロックごとの書換えでは、ブロック内の個々のデータの書換え回数としては上限値を越えないにも係わらず、ブロック全体として書き換えるために上限値を越えてしまうことがあって、効率的でないことか

ら、1データごとに書換え回数記録用のカウンタが設けられるメモリ技術もある。

**【0003】**

**【発明が解決しようとする課題】** しかしながら、上記のいずれの書換えのシステムにおいても、関連のある複数のデータがメモリの一領域に書き込まれる場合に、その一連のデータの書換えが完了する前に給電が遮断されたような時には、結果としてその一領域においてデータが変更された部分と変更されていない部分とが存在する。このため、再度通電された際に、その領域の一連のデータを読出しても、それらのデータは正確ではないという問題がある。また、データの書込み途中での給電の遮断と判定されたとしても、常時同じ領域にデータを書き込んでいるため、適正データのうち最後に記憶されたデータも読みだすことができない。また特に、後者の書換えのシステムでは、必ず書換え回数まで利用できるものの、メモリの半分がカウンタで占められることになり、メモリとしての利用効率が悪いという問題がある。

**【0004】** 従って上記の課題に鑑み、本発明では、書込み中断などの不具合が発生しても、前回の正常な記憶動作の完了したデータを保持し読出し出来るようにすることを目的とする。

**【0005】**

**【課題を解決するための手段】** 上記の課題を解決するため本発明の構成は、書換え制限回数を有する不揮発性のメモリを有して読出し書込みがプログラムで実施されるメモリ装置において、一回のデータの読出しと書込みの一単位となる複数のブロックに分割されたデータ記憶手段と、前記ブロック一つ一つに対して、書込終了時に又は書込開始時にその旨の状態を示す参照データの少なくともいずれか一つを記憶する状態記憶手段と、前記参照データを用いて、最後にデータの書込みが正常終了したブロックを読出可能ブロック、その次のブロックを書込可能ブロックと判定するブロック判定手段と、前記読出可能ブロックに対してデータを読出し、または前記書込可能ブロックに対してデータを書込むと共に、前記状態記憶手段において前記書込開始時又は書込終了時に、その参照データを更新する読出書込手段とを有し、前記読出可能ブロックまたは前記書込可能ブロックを順次更新することである。また関連発明の構成は、前記状態記憶手段は、書込開始状態と書込終了状態とをそれぞれ示す参照データを記憶し、前記読出書込手段は、書込開始時に書込み開始を示す前記参照データと、書込み終了時に書込終了を示す前記参照データの更新を行うことである。

**【0006】**

**【作用】** 書換え回数が制限された不揮発性のメモリ、例えばEEPROMの記憶領域を複数のブロックに分割して、必要とするデータの書込みが、順に前回と異なって選択されたブロックに対して実施される。読出可能ブロックと

書込可能ブロックの選択は、各ブロック毎に対応した書込み終了または書込み開始の状態を示す、状態記憶手段に記憶された参照データを参照することで行われる。

【0007】状態記憶手段に書込み終了の状態のみが記憶されている場合は、ブロック判定手段で参照データを順に調べ、前回書込みが完了したブロックが判定され、そのブロックが読出可能ブロック、その次のブロックが書込可能ブロックと決定される。初回の書込みはどのブロックも書込まれていないので、一番めのブロックが対象となる。データの書込みが完了すると、その書込みを行ったブロックに対応した状態記憶手段に書込み終了を意味する参照データ値を記憶させる。データ書換え途中に給電が遮断されて書換え処理が停止し、不完全にデータ書換えが終了したとしても、その後再び電源が回復して、書換えたデータを参照する際に、書換え中断のブロックは書込み終了の参照データが設定されていないので、その前のブロックが最新データを記憶していると認識される。

【0008】状態記憶手段に書込み開始の状態のみが記憶されている場合は、ブロック判定手段で参照データを順に調べ、前回書込みが開始されたブロックが判定され、このブロックが書込可能ブロック、その前のブロックが読出可能ブロックと決定される。この場合はデータ書換え途中に給電が遮断されて書換え処理が停止されると、その後再び電源が回復して、書換えたデータを参照する際に書換え中断の判定はできないため、その前の前のブロックが確実なデータを記憶していると認識される。

【0009】請求項2の構成では、状態記憶手段に、書込み終了と書込み開始の状態が記憶されている場合であり、ブロック判定手段で、書込み終了の状態を調べて前回書込みが完了したブロックが判定され、そのブロックが読出可能ブロック、その次のブロックが書込可能ブロックと決定される。さらに、書込み終了となっているブロックと書込み開始となっているブロックとの状態が比較されて、前回書込みが中断されたか完了したかが判定される。データ書換え途中で電源が遮断されて書換え処理が停止し、不完全にデータ書換えが終了したとしても、その後再び電源が回復して、書換えたデータを参照する際に、書換え中断のブロックは書込み終了の参照データが設定されていないので、その前のブロックが最新データを記憶していると認識される。

【0010】

【発明の効果】書換え回数が制限された不揮発性のメモリ、例えばEEPROMの記憶領域をブロックに分割して使用する構成とし、書込みをブロックごとに実施することで書換え回数が見かけ上増大される。また書込み終了または書込み開始の少なくともいずれかを示す参照データを設けて参照するので、途中で書込みが中断して終了した場合でも、再開後に中断以前に最後に適正に記憶された

データを利用できる。請求項2の構成では、書込みを実施するブロックが決定されるだけでなく、そのブロックの現状として、中断状態で終了したために書込可能なのか、前のブロックが正常に書き込まれて終了した状態で書込可能なのかが判別でき、重複した書込みを避けたり、コードの照合などに不具合が発生しないように処理をとらせることができる。

【0011】

【実施例】以下、本発明を具体的な実施例に基づいて説明する。図1は、内燃機関制御装置1と車両盗難防止装置9との間で共有するコード（ローリングコードとも呼ばれる）の記憶に本発明のメモリ装置を用いた内燃機関制御システム100のブロック構成図である。なお図1では、書換え回数が制限されたメモリであるEEPROMをE<sup>2</sup>PR<sup>2</sup>OMと表示している。

【0012】内燃機関制御装置1はマイクロコンピュータで構成される。CPU2には、内燃機関（図示しない）に設けられている回転センサ3からの回転信号S1を入力する波形整形回路5が接続され、また内燃機関の吸気管に設けられた吸気圧センサ4による吸気管内圧力信号S2を入力するフィルタ6が接続されている。

【0013】CPU2と接続されているプログラムメモリ（ROM）7には制御プログラムが記憶されており、この制御プログラムをCPU2が実行することで内燃機関の制御が実行される。前記の各信号S1、S2が制御プログラムで使用され、ここではSPI（Single Point Injection）機能として内燃機関のインジェクタ8が制御される。その他図示しないが、CPU2に接続されたA/D変換器やD/A変換を制御して、制御に必要となる信号がやり取りされている。

【0014】内燃機関制御装置1のCPU2は、車両盗難防止装置9と接続されている。車両盗難防止装置9は、内燃機関が、正常な始動操作により始動されたかどうかを監視するものである。車両盗難防止装置9からは、内燃機関の始動のつど、前もって始動時の正常な始動操作に対応して装置内に記憶されていたコードが内燃機関制御装置1に送信される。そのコードは内燃機関制御装置1内のCPU2に接続されているEEPROM10にも、前もってその他の必要な情報とともに一群のデータとして記憶されている。EEPROM10は、後述するように内部がN個のブロックに分割されて使用され、参照データが記憶される状態記憶手段として書込み開始テーブル11と書込み終了テーブル12がEEPROM10内に設けられている。

【0015】内燃機関制御装置1は、車両盗難防止装置9から送られてきたコードをEEPROM10に記憶しているコードと比較し、これらが一致しない場合は異常操作による内燃機関の始動と判定し、インジェクタ8の駆動を停止させる。コードが一致した場合は正常な始動操作と判断されるので、一致したことが車両盗難防止装置9に

通知される。そして再び車両盗難防止装置9から、内燃機関の始動操作に対応した以前と異なる新しいコードが内燃機関制御装置1に送信されて、その他必要な情報と共に一群のデータとしてEEPROM10に記憶される。このようにEEPROM10は始動操作が正常に行われる度ごとに書換えられる（読込書込手段）。なお、車両盗難防止装置9は、書換え時のフェールセーフ機能として、一回前のコードも保有し、照合させるために送出する機能がある。

【0016】EEPROM10への書込みは次のようになっている。EEPROM10自体の物理的限界から、通常10000回程度の書換え制限回数が記憶データの保証の観点から設定されている。一方、車両盗難防止装置として車両の運用期間、例えば10年もしくは15年間に渡るコードの最大の書換え回数を想定し、これを最大使用回数 $N_{\max}$ とすると、EEPROMのコード記憶領域を下記1式で示されるNブロックに分割し、これらブロックに順次、かつ循環的にコードを書換え記憶させることで、EEPROMの書換え制限回数を越える可能性を低くすることができる。

【数 1】 $N_{\max} / 10000 \leq N$

【0017】なお、EEPROM10への書込みや読出しは1ブロック単位で実施される（データ記憶手段）。そして、前回の最新データ（コードを含む情報）を上書きしないように、1回の書込み毎に、書込みを行うブロックが順次変更されていく。従って書込みが中断した場合でも、それ以前の前回書込み完了した最新データが必ず残されている。

【0018】N個のブロックそれぞれに対する書込みが正常に終了したか、また書込みが開始されたかどうかその後で確認されるように、書込開始時と書込終了時にそれらの状態を示す参照データを各ブロック個々に対応させて記憶させる書込み開始テーブル11と書込み終了テーブル12とが用意される。この書込み開始テーブル11と書込み終了テーブル12に記憶されるテーブル値（参照データ）は電源オフ後にも記憶されている必要があることからEEPROM10に記憶される。書込み開始テーブル11及び書込み終了テーブル12は、図3に示すようなN個分の値が記憶できるテーブルである。それぞれのテーブルの各アドレスに記憶されている値は、そのアドレスに対応するブロックの書込開始状態および書込終了状態を示している。従ってこの各テーブル11、12は必ずしも連続したメモリ領域に取られる必要はなく、EEPROM10の各ブロック領域の一部に属するように設けられてもよい。またはEEPROM10の一區画にこのテーブル領域を設けて、残りの領域をN個のブロックに分割するようにしても良い。

【0019】これらのテーブル値をCPU2が調べること、最新データが記憶されている読出可能ブロックと次に書込むべき書込可能ブロックとが判定される（ブロック判定手段）。なおこのCPU2のブロック判定手段

は、プログラムメモリ7に記憶されている制御プログラムに含まれている。

【0020】今ここで、EEPROM10の記憶領域のN個分（数1式を満たすN）のブロックを、1番目からN番目まで順に書換えが実施されていくものとする。そして、N個のブロックの内のX番目まで書込みが完了しており、このX番目のブロックから一群のデータが読出されてコードが読みだされ、然るべき盗難防止判定の後に、次のX+1番目のブロックに新たなコードを含んだ一群のデータが書き込まれるとする。X+1番目のブロックに書込まれる前の段階では、書込み開始テーブル11には、図3(a)に示すように、1番目からX番目の各ブロックに対してはすでに書込みが実施されているので、書込み開始が既に行われたことを示すY+1というテーブル値が記憶されている。そしてX+1番目以降のブロックに対するテーブル値はYという値が記憶されていて、まだ書込みが実施されていないことを意味している。つまり書込み開始テーブル11の記憶内容は、書込みとしてそのブロックが使用された場合に、現在記憶している値（ここではY）に1を加えた値が、書込みを始める際に書換えられる。

【0021】上記のような場合で、内燃機関制御装置1が始動時に車両盗難防止装置9からコードを受信した時、EEPROM10に記憶している前回の最新データからコードを読出し、キー操作の正常異常判定を行い、さらに適正始動の場合に新しいコードを受信してEEPROM10へ新しいデータを記憶する手順を図2のフローチャートを用いて説明する。

【0022】(1) まずステップ202で、読出しを実施すべき読出可能ブロックが何番目のブロックであるかが調べられる。読出可能ブロックが決定されれば、次に書込みをする書込可能ブロックはその番号+1番目のブロックとなる。書込み終了テーブル12が参照され、そのテーブル値を順に検索し、隣接するテーブル値が比較される。それらの値が異なる値となったX+1番目のブロックがまだ書き換えられていないブロック、即ち書込可能ブロックと決定され、X番目のブロックが、前回書換えられて最新データが残されているブロック、即ち読出可能ブロックと決定される。ここでは最後に書き込まれたX番目のブロックに対応するテーブル値にY+1が書き込まれており、その次のX+1番目のブロックに対応するテーブル値にはYが書き込まれている（図3(a)）。

【0023】(2) ステップ204で、読出すべきX番目のブロックに記憶されているデータを読出す処理が実施され、また車両盗難防止装置9から始動操作によって発生するコードが送られてくるのを受信処理する。そしてステップ206で、記憶されていたコードと車両盗難防止装置9から送られてきたコードとが比較されて盗難判定が実施される。

(3) ステップ206のコード比較の結果、コードが不一致

であれば異常始動であるとされるので、新しいコードを受け取って記憶しなおす必要がなく、別の処理を実施するために、このフローを終了する。コードが一致していれば、正常な始動であるとされ、次のステップに進む。なお、実際にはこのコード比較には、書込みが中断された場合のコード比較も含まれる。

(4) ステップ208 で、まず正常であることが車両盗難防止装置9に通知される。そして車両盗難防止装置9から、新しいコードが内燃機関制御装置1に対して送出されるので、その新しいコードを受信処理する。

【0024】(5) 内燃機関制御装置1では、新しいコードに基づくデータの書込みを実施するため、ステップ210で、書込み開始テーブル11のX+1番目のブロックのテーブル値YがY+1に更新される(図3(b))。そしてステップ212で、X+1番目のブロックに対して、車両盗難防止装置9から送られてきたその新しいコードによるデータが書込まれる。

(6) そのデータ処理が終了したら、ステップ214で、書込み終了テーブル12のX+1番目のテーブル値をYからY+1に更新して(図3(c))、書換えが完了したことの情報をセットして書換の処理を完了する。

【0025】またステップ212の最中に、書込みが何らかの理由で途中で停止し、正常に終了しなかった場合には、X+1番目のブロックに書き込まれているデータに含まれるコードは信頼性がなく、従ってこのX+1番目のデータは不要であり、再度X+1番目のブロックに書き込まれても構わない。前回の書込みが途中で終了しているかどうかを確認するためには、書込み開始テーブル11を調べる。X+1番目のテーブル値がYとなっているので、このX+1番目のブロックは途中で中止されたものではないことが確認される。

【0026】中断後、再びプログラムが再開され、ステップ202で読出可能ブロックと書込可能ブロックが判定される際に、図3(c)における書込み終了テーブル12のX+1番目のメモリがYのままとなっている。書込みが終了された完全な最新データはX番目のブロックに記憶されているので、X番目のブロックが読出可能ブロック、X+1番目のブロックが書込可能ブロックと判定される。中断されて更新されたコードが正しく記憶されていないという場合は、上述のステップ206の内部で、まずコードが一致せず、中断か否かの判定が成されて、中断であれば、その旨が車両盗難防止装置9に通知されて、一回前のコードを呼出させて、最新データに含まれる一回前のコードとコード比較を行うことで対応できる。従って途中で処理が中断したような場合でも、なんらかの後処理を必要とせず、必要とする正しいデータが読出され、問題なくもう一度途中で書込みが止まったブロックに対して新しいデータの書込みが実施される。

【0027】この書込みが中断して、再度同じブロックに書込みが実施されると、二重に書込みが実施されたこ

とになる。同じブロックで何回か同じ現象が発生すると、いつかは実質の書込みが書換え制限回数である10000回を越えることになる。しかし上記のような特殊な書き直し状況が同じブロックで多発する確率は少なく、各ブロックは順番に書換えられていくので平均して書換え回数が増大していく。従ってEEPROM10の書込み性能に対する影響はほとんどない。また予めシミュレーションなどで予想される中断発生回数を見込んでおき、書換え制限回数から差し引いておいてもよい。

【0028】しかし書込み開始テーブル11を参照することによって、書込み終了テーブル12の参照で判定された次の書込可能ブロックが、中断で終わったのか、前のブロックが正常に完了して全く書込みが開始されていないのかを確認することができる。それで、中断であると判定された場合、書込み回数を1回増やさないように次のX+2番目のブロックを書込可能ブロックとすることができる。ただし、読出可能ブロックはX番目のブロックである。

【0029】参照データとなる書込み開始および書込み終了の各テーブル11、12において、1～Nまで順次、各ブロックに対するテーブル値を書換え毎に同じ値(図3ではY+1)に設定していくことにすれば、常に最新データがどのブロックに存在するのかを、テーブル値の変化のある境界で判定でき、その前後が読出可能ブロック、書込可能ブロックであることが容易に判定される。このような方式で検索を実施することから、テーブル値は歩進する数値で構成される必要はなく、書込まれたブロックとこれから書込まれるブロックとの境界が判定されるような値で良く、それには少なくとも1ビット有れば良い。

【0030】N個のブロックそれぞれが書換え制限回数を越えないようにするためには、N番目のブロックが書き換えられる時点ごとに、別に設けておいた一つのカウンタを1ずつインクリメントさせていき、このカウンタの値が10000を越えるかどうかを調べることでできる。どのブロックも順番に書換えられるのでいずれかのブロックの書換え回数を調べれば良いからである。あるいは、前述の各テーブル値をインクリメントさせていくカウンタ値とさせても良い。

【0031】以上のように、実際の書込み回数をEEPROM10の書換え制限回数を越えて使用できるようにしたメモリ装置でも、特別な不具合対策の手段を設けることなく、前回書込んだデータを確実に保持して再度読みだすことができる。

【0032】なお、書込まれたブロックがちょうどN番目であった場合には、次に書き込まれるべきテーブルのメモリは1番目に戻るようになる。N番目のブロックに対して書込みが終了した時点での各テーブル11、12の状態は図4(a)に示すとおりである。図4(a)ではテーブル11、12の総ての値が同一のY+1になってお

り、総てのテーブルのデータ値が変化なしとなる。データ値に変化があることを検出して読出可能ブロックと書込可能ブロックとを決定しているアルゴリズムでは、その状態ではそれらのブロックが決定されない。従ってこのような場合は、ちょうど総てのメモリが一通り書き込まれた状態なので、再び1番目から書込みが実施されていく処置をとることとする。

【0033】そして、1番目のブロックに書込みを実施する場合、書込み開始テーブル11には、書込み前の値 $Y+1$ に1を加えた $Y+2$ が書き込まれ(図4(b))、1番目のブロックに新しいコードが書き込まれた後に、書込み終了テーブル12の1番目の所に、同様に $Y+2$ が書き込まれる(図4(c))。なお、この $Y+2$ が $Y$ であっても良い。これはテーブルのメモリが1ビットであれば、前の値に+1を実施すると自動的に0と1の値になることに対応している。従って $N$ 個分のテーブルのために、少なくとも $2N$ ビット分用意するだけで済み、EEPROMのメモリ領域を多く専有してしまわない。

【0034】(第二実施例) 第一実施例の場合は、EEPROM 10に書込み開始テーブル11と書込み終了テーブル12の二つが設置された場合を示したが、このうち書込み終了テーブル12が設置されるだけの場合について説明する。書込み終了テーブル12だけが設置されている場合には、書込み終了テーブル12のテーブル値が変化している境界が、書込みが完了して最新データが記憶されているブロック、即ち読出可能ブロックと、これから書込まれるべきブロック、即ち書込可能ブロックとの境

界を示すことになり、かつ書込みが無事完了していることも判る。この場合にはそのブロックが書込みが開始されていたかどうかは調べることができない。つまり中断であったか、前のブロックが正常に書込み終了しただけなのかの区別はつかない。この場合でも中断が発生した後の処理は第一実施例と同じことになるので問題はない。この第二実施例の場合には、EEPROM 10に占める領域が少なくとも $N$ ビット分だけでよい利点があるため、従来技術にあるようなカウンタをメモリごとに設ける場合に比べてメモリの利用効率率は格段に向上する。

【図面の簡単な説明】

【図1】本発明のメモリ装置を利用する内燃機関制御装置のブロック構成図。

【図2】本発明のメモリ装置に対する読出し、書込み処理のフローチャート。

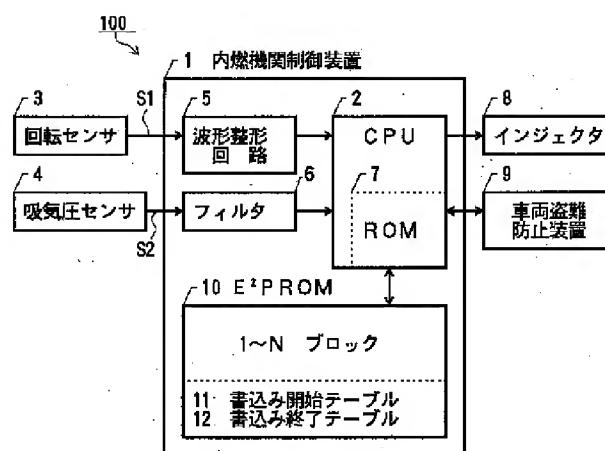
【図3】書込み開始テーブルおよび書込み終了テーブルの説明図。

【図4】書込み処理の特殊な例の説明図。

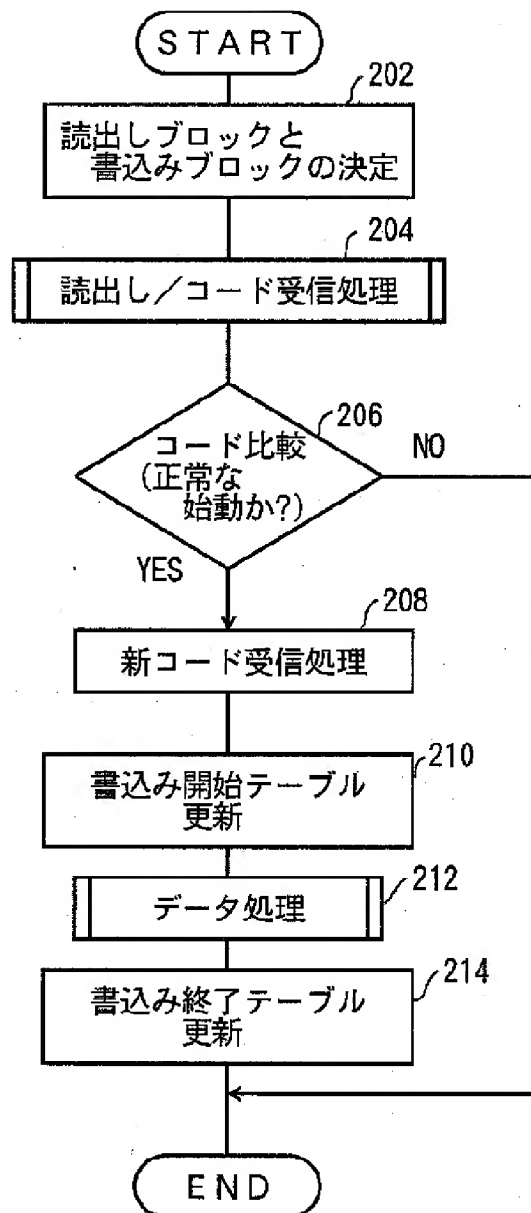
【符号の説明】

- 1 内燃機関制御装置
- 7 ROM (制御プログラム、状態記憶手段、ブロック判定手段を含む)
- 9 車両盗難防止装置
- 10 EEPROM (読出可能ブロック、書込可能ブロックを含む)
- 11 書込み開始テーブル (参照データ)
- 12 書込み終了テーブル (参照データ)

【図1】

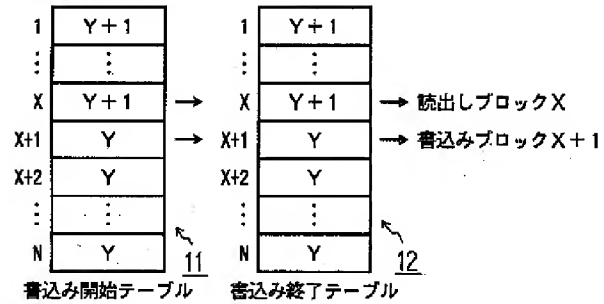


【図2】

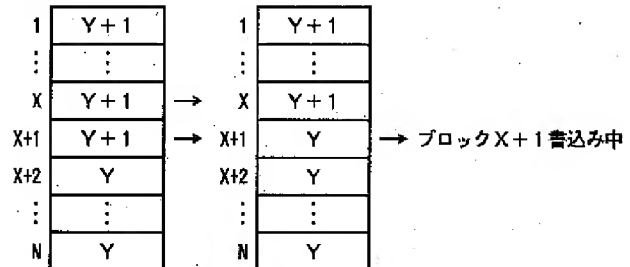


【図3】

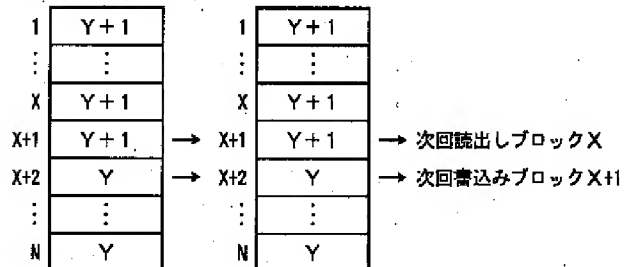
(a) 書込み開始前1



(b) 書込み中1

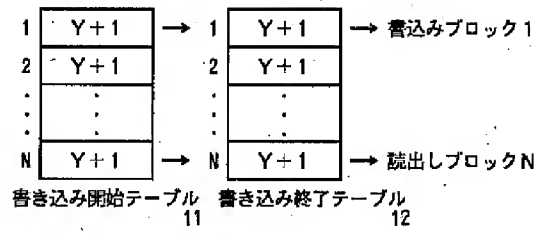


(c) 書込み終了1

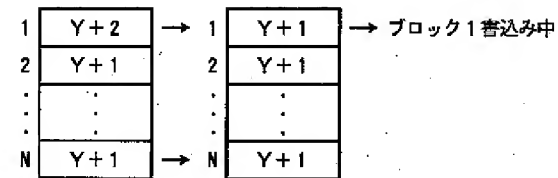


【図 4】

(a) 書き込み開始前 2



(b) 書き込み中 2



(c) 書き込み終了 2

